## CLAIMS

1. In a network-connected integrated circuit, a method for securely buffering overhead messages, the method comprising:

receiving messages including overhead bytes;

collecting overhead bytes;

creating a first overhead message from the collected overhead bytes; and,

saving the first overhead message until it is read.

2. The method of claim 1 wherein receiving messages including overhead bytes includes receiving messages in a frame format;

wherein collecting overhead bytes includes collecting a first number of overhead bytes per frame from a second number of frames;

wherein creating a first overhead message from the collected overhead bytes includes writing the first number of collected bytes from each of the second number of frames to a buffer; and,

wherein saving the first overhead message until it is read includes not overwriting the first overhead message stored in the buffer until the buffer is read.

3. The method of claim 2 further comprising:

reading the first overhead message in the buffer;

collecting new overhead bytes;

creating a second overhead message from the collected new overhead bytes; and,

saving the second overhead message until it is read.

4. The method of claim 3 further comprising:

establishing an overhead message semaphore; and,

wherein saving the first overhead message until it is read includes overwriting the buffer with the second overhead message in response to the semaphore.

5. The method of claim 4 wherein not overwriting the first overhead message stored in the buffer until the buffer is read includes:

setting the semaphore to the lock state; and,

in response the semaphore lock state, ceasing the writing of collected overhead bytes to the buffer for the second overhead message.

6. The method of claim 5 wherein collecting overhead bytes for the second overhead message includes:

setting the semaphore to the unlock state following the reading of the first overhead message;

collecting a first number of overhead bytes from a second number of frames;

wherein creating a second overhead message from the collected new overhead bytes includes writing a first number of new overhead bytes to the buffer in response to the semaphore unlock state; and,

wherein saving the second overhead message until it is read includes setting the semaphore to the lock state in response to creating the second overhead message in the buffer.

7. The method of claim 6 wherein receiving a message including overhead bytes includes receiving a G.709 format message with 16 overhead bytes per row.

8. The method of claim 7 wherein collecting a first number of overhead bytes per frame from a second number of frames includes collecting trail trace identifier (TTI) bytes every frame, for 64 frames; and,

wherein creating a first overhead message includes creating an overhead message from the TTI bytes in the 64 frames.

9. The method of claim 7 wherein collecting a first number of overhead bytes per frame for a second number of frames includes collecting messages selected from the group including fault type and fault location (FTFL), general communication channel (GCC), experimental (EXP), and automatic protection switching/protection communication control (APS/PCC) messages.

10. The method of claim 7 wherein establishing a overhead message semaphore includes establishing a byte semaphore for each overhead byte in the buffer; and,

wherein saving the first overhead message until it is read includes overwriting each overhead byte in the first overhead message with an overhead byte in the second overhead message in response to a corresponding the byte semaphore.

11. In a network including a data processor, a method for securely buffering overhead messages, the method comprising:

receiving a message including overhead bytes at a data processor;

collecting overhead bytes;

creating a first overhead message from the collected overhead bytes; and,

saving the first overhead message until it can be read by a microprocessor.

12. The method of claim 11 wherein receiving a message including overhead bytes at a data processor includes receiving messages in a frame format;

wherein collecting overhead bytes includes collecting a first number of overhead bytes per frame from a second number of frames;

wherein creating a first overhead message from the collected overhead bytes includes writing the first number of collected bytes from each of the second number of frames to a buffer; and,

wherein saving the first overhead message until it is read includes not overwriting the first overhead message stored in the buffer until the buffer is read.

13. The method of claim 12 further comprising:

the microprocessor reading the first overhead message from the buffer;

the data processor collecting new overhead bytes;

the data processor creating a second overhead message from the collected new overhead bytes; and,

the data processor saving the second overhead message until it is read.

14. The method of claim 13 further comprising:

establishing an overhead message semaphore; and,

wherein saving the first overhead message until it is read includes the data processor overwriting the buffer with the second overhead message in response to the semaphore.

15. The method of claim 14 wherein not overwriting the first overhead message stored in the buffer until the buffer is read includes:

the data processor setting the semaphore to the lock state; and,

in response the semaphore lock state, the data processor ceasing the writing of collected overhead bytes to the buffer for the second overhead message.

16. The method of claim 15 wherein collecting overhead bytes for the second overhead message includes:

the microprocessor setting the semaphore to the unlock state following the reading of the first overhead message;

the data processor collecting a first number of overhead bytes from a second number of frames;

wherein creating a second overhead message from the collected new overhead bytes includes the data processor writing a first number of new overhead bytes to the buffer in response to the semaphore unlock state; and,

wherein saving the second overhead message until it is read includes the data processor setting the semaphore to the lock state in response to creating the second overhead message in the buffer.

17. The method of claim 16 wherein receiving a message including overhead bytes at the data processor includes receiving a G.709 format message with 16 overhead bytes per frame.

18. The method of claim 17 wherein collecting a first number of overhead bytes per frame for a second number of frames includes the data processor collecting trail trace identifier (TTI) bytes every frame, for 64 frames; and,

wherein creating a first overhead message includes creating an overhead message of the TTI bytes in the 64 frames.

19. The method of claim 17 wherein collecting a first number of overhead bytes per frame for a second number of frames includes collecting messages selected from the group including fault type and fault location (FTFL), general communication channel (GCC), experimental (EXP), and automatic protection switching/protection communication control (APS/PCC) messages.

20. The method of claim 17 wherein establishing a overhead message semaphore includes establishing a byte semaphore for each overhead byte in the buffer; and,

wherein saving the first overhead message until it is read includes the data processor overwriting each overhead byte in the first overhead message with an overhead byte in the second overhead message in response to a corresponding the byte semaphore.

21. In a network-connected integrated circuit, a system for securely buffering overhead messages, the system comprising:

a processor having an input for receiving messages including overhead bytes and an output for supplying overhead bytes;

a message buffer having an input to accept overhead bytes, the buffer collecting overhead bytes to create an overhead message from the collected overhead bytes and supplying the overhead message at an output; and,

a semaphore register to protect the buffered overhead message from being overwritten until the buffer is read.

22. The system of claim 21 wherein the processor receives messages in a frame format and supplies a first number of overhead bytes per frame for a second number of frames;

wherein the semaphore register protects the buffered overhead message in the message buffer by alerting the processor that the overhead message has not yet been read; and,

wherein the processor supplies overhead bytes to the buffer in response to the semaphore register.

23. The system of claim 22 wherein the semaphore register has an input to accept a lock value; and,

wherein the processor ceases to supply overhead bytes to the buffer in response to the lock value loaded in the semaphore register.

24. The system of claim 23 wherein the semaphore register input accepts an unlock value; and,

wherein the processor supplies overhead bytes to the buffer in response to an unlock value loaded in the semaphore register.

25. The system of claim 24 wherein the processor receives a G.709 format message with 64 overhead bytes per frame.

26. The system of claim 24 wherein the processor supplies trail trace identifier (TTI) overhead bytes every frame, for 64 frames; and,

wherein the message buffer accepts a trail trace identifier (TTI) byte every frame, creating an overhead message from storing collected TTI bytes from the 64 frames.

27. The system of claim 24 wherein the processor supplies bytes selected from the group including fault type and fault location (FTFL), general communication channel (GCC), experimental (EXP), and automatic protection switching/protection communication control (APS/PCC) messages for one frame; and,

wherein the message buffer accepts a bytes from the selected group every frame, creating an overhead message.

28. The system of claim 24 wherein the semaphore register includes a second number of semaphore registers corresponding to each one of the first number of overhead bytes collected in the second number of frames;

wherein the processor loads the unlock value into each semaphore register in response to writing the corresponding second number of overhead messages into a second number of memory locations in the buffer; and,

wherein an unlock value is loaded into each semaphore register in response to reading the corresponding second number of overhead messages from the buffer.

29. In a network of connected processors, a system for securely buffering overhead messages, the system comprising:

a data processor having an input for receiving messages including overhead bytes, an output for supplying overhead bytes, and an output to supply semaphore lock values;

a message buffer having an input to accept the overhead bytes, the buffer collecting overhead bytes to create an overhead message from the collected overhead bytes and supplying the overhead message at an output;

a semaphore register to protect the overhead message in the buffer from being overwritten; and,

a microprocessor having an input to read the overhead message in the buffer.

30. The system of claim 29 wherein the semaphore register has an input to accept lock and unlock values; and,

wherein the microprocessor has an output to change the lock value in the semaphore register to the unlock value, in response to reading the buffer.

31. The system of claim 30 wherein the data processor ceases to supply overhead bytes to the buffer in response to the lock value being loaded in the semaphore register.

32. The system of claim 31 wherein the data processor receives messages in a frame format and supplies a first number of overhead bytes per frame, for a second number of frames.

33. The system of claim 32 wherein the data processor receives a G.709 format message with 16 overhead bytes per row.

34. The system of claim 33 wherein the data processor supplies trail trace identifier (TTI) overhead bytes every frame, for 64 frames; and,

wherein the message buffer accepts trail trace identifier (TTI) bytes every frame and creates an overhead message from storing the collected TTI bytes from the 64 frames.

35. The system of claim 33 wherein the processor supplies bytes selected from the group including fault type and fault location (FTFL), general communication channel (GCC), experimental (EXP), and automatic protection switching/protection communication control (APS/PCC) messages every frame; and,

wherein the message buffer accepts a bytes from the selected group every frame, creating an overhead message.

36. The system of claim 33 wherein the microprocessor reads the overhead message from the buffer and transmits the overhead message in an upstream message.

37. The system of claim 32 wherein the semaphore register includes a second number of semaphore registers corresponding to each one of the first number of overhead bytes collected in the second number of frames;

wherein the data processor loads the lock value into each semaphore register in response to writing the corresponding second number of overhead messages into a second number of memory locations in the buffer; and,

wherein the microprocessor loads the unlock value into each semaphore register in response to reading the corresponding overhead message from the buffer.